

I. Cross Reference to Related Application

10 **II. Field of the Invention**

15

20

25

30

to redirect the customer to these different servers without requiring the customer to sign on each and every time he or she goes accesses a different server.

Conventional products attempting to address this problem are deficient, for example, both in terms of performance and cost. In some such products, it is necessary to return to a centralized resource. Other such products do not support crossing organizational boundaries or Internet domain boundaries. What is needed is a methods and system for single sign-on user access to multiple web servers, such as a federation of web servers sharing sub-domains, that overcome such disadvantages and that provide other advantages.

10

IV. Summary of the Invention

The present invention provides methods and systems for single sign-on user access to multiple web servers, such as a federation of web servers sharing sub-domains, are provided. In an embodiment, a user is authenticated at a first web server (e.g., by user name and password). The first web server provides a web page to the user having a service selector (e.g., a hyperlink comprising the URL of a second web server offering the service indicated by the selector). Upon activation of the selector by the user, the first web server constructs, digitally signs, and transmits an encrypted authentication token (e.g., a cookie) from the first web server to a second web server via the user client. URL encoding is employed to encrypt and sign the authentication token. The first and second web servers share a sub-domain. The authentication token comprises an expiration time. The authentication token is examined and authenticated at the second web server and URL decoding is employed. Upon authentication, and if the expiration time has not passed, the second web server allows the user to conduct a session at the second web server.

In another embodiment, a method for single sign-on user access to a federation of web servers, such as a first and second web server sharing a sub-domain, is provided. In an embodiment, the method comprises allowing a user at a computing device to access a first web server in the federation of web servers via a web browser of the computing device, authenticating the user with user-provided authentication information, including at least a user identification, by the first web

server, and allowing the user to conduct a session a the first web server. During the session, the first web server carries out the functions of prompting the user for selection of a functionality offered via at least a second web server, and receiving a selection by the user of the functionality offered via the second web server.

- 5 Upon receiving a selection, the first web server creates an authentication token for the user including at least the user identification and with a pre-defined token expiry by the first web server, digitally signing (e.g., by public key encryption) the authentication token by the first web server. An embodiment further comprises qualifying the domain attribute of the authentication token with the shared sub-
- 10 domain name by the first web server, sending the digitally signed authentication token to the web browser of the computing device by the first web server, redirecting the web browser to the second web server by the first web server, and sending the authentication token to the second web server by the web browser. The second web server decrypts the authentication token, confirms a match with
- 15 the digital signature of the first web server, checks the pre-defined expiry of the authentication token by the second web server, and allows the user to conduct a session with the second web server if within the pre-defined token expiry.

- In another embodiment, a method of single sign-on for multiple web servers, comprising receiving log-in data from a user in a first server, providing
- 20 the user with a service selector, receiving an indication that the user selected the service selector, constructing an authentication token comprising profile data associated with the user, encrypting and signing the authentication token, redirecting the user to a second server, transmitting the authentication token to the user, receiving the authentication token in the second server, verifying the
 - 25 authentication token in the second server, and allowing the user access to a service provided by the second server. In one such embodiment, the authentication token comprises a cookie and further comprises expiration time data.

- It is a feature and advantage of the present invention to provide a method and system for single sign-on user access to a federation of web servers that allows
- 30 users already authenticated on a web site, such as a financial institution's web site, to have access, for example, to a service provider's web site without having to be re-authenticated via provision of a valid name and password.

To achieve the stated and other features and advantages, an embodiment of the present invention provides a method and system which enables single sign-on access for a user to a federation of web servers that allows user authentication at an entity's web site server, selection of a service provider URL, and passage of an authentication token by the entity's web site server to a service provider's server that contains sufficient information to enable the service provider's server, for example, to recognize the user as a valid service provider user and to provide the user with customer specific information.

Additional objects, advantages and novel features of the invention will be set forth in part in the description which follows, and in part will become more apparent to those skilled in the art upon examination of the following, or may be learned by practice of the invention.

V. Brief Description of Figures

FIG. 1 shows an embodiment of a system according to the present invention.

FIG. 2 shows a flow diagram describing a process according to the present invention carried out in the system shown in FIG. 1.

FIG. 3 shows a visual depiction of web pages associated with web servers of the system shown in FIG. 1.

FIG. 4 shows a flow diagram describing an alternative process according to the present invention carried out in the system shown in FIG. 1.

VI. Detailed Description

FIG. 1 shows an embodiment of a system according to the present invention. A brokerage firm web server (BFWS) 30 includes a brokerage firm web site 32. The brokerage firm web server 30 (and likewise the brokerage firm web site 32) is in communication with the Internet 20. Likewise, a bank web server 40 includes a bank web site 42. The bank web server (BWS) 40 (and likewise the bank web site 42) is also in communication with the Internet 20. A customer 5 of both the brokerage firm and the bank is shown. The customer 5 has a user name and password to access both the brokerage firm web site 32 and the

bank web site 42. The customer's personal computer 10 having a web browser such as Microsoft Internet Explorer or Netscape Navigator (a client) is in communication with the Internet 20 as well. The customer 5 uses the customer's personal computer and browser 10 to communicate with the web servers 30, 40 via the Internet. Herein, the term customer is used in many instances to indicate the client 10 as used by the customer. In addition to network communication facilities and other aspects, the brokerage firm web server 30 and the bank server 40 comprise programming to carry out the functions described herein.

FIG. 2 shows a flow diagram of steps carried out in the system shown in FIG. 1. The customer 10 points the customer's browser to the brokerage firm web site 32. The customer 10 logs into the brokerage firm web site 32 using the customer's correct user name (or user identification) and password for the brokerage firm web site 32, and is authenticated by the brokerage firm web server 30. Once the customer logs into the brokerage firm web site 32, the web site 32 presents the customer 10 with a welcome page from the web site 32. Once logged in, the customer 10 may examine the customer's brokerage account information, portfolio, investment information, and the like.

FIG. 3 shows a graphical depiction of the system shown in FIG. 1, including the welcome page 100 from the brokerage firm web site 30 provided to the customer 10 once the customer logs in at the brokerage firm web site 30. The welcome page 100 shown in FIG. 3 includes a service selector in the form of a hyperlink shown as "bill payment" 102. The brokerage web site offers its customers the ability to pay bills.

Referring again to FIG. 2, the customer 10 requests bill payment 50 by clicking on the "bill payment" hyperlink 102. The brokerage firm web server 30 itself does not handle the process of bill payment, but the server 30 is programmed with the knowledge that the bank web server 40 handles such a process. The hyperlink 102 includes the URL of the bank web site 42. Upon detecting the request of bill payment, the brokerage firm server 30 builds an authentication token 52. An authentication token comprises an object (or data) that can be passed between cooperating servers. A function of an embodiment of an authentication token is to convey the necessary information from a primary (or first) server to a

secondary (or second) server to allow the secondary server to skip the sign-on process that would otherwise be necessary and required. Once a primary server establishes a session for a user, a cooperating secondary server that receives a valid authentication token from the primary server can establish a session without
5 having the user sign on again.

In the embodiment shown in FIG. 1, the brokerage firm web server 30 builds an authentication token (or access token) comprising user identification data (or profile data) and expiration time data (token expiry) 52. The profile data comprises user identification data comprising a customer identification number
10 that uniquely identifies the user to the secondary server. In the shown embodiment, the token also include a list of accounts of the customer. Expiration time data comprises data reflecting the time after which the authentication token is invalid. In the embodiment shown, the time is in Greenwich Mean Time (GMT). In other embodiments, the time may be in Universal Time. Expiration time may
15 be set by the primary server at any desired time, though in most embodiments the expiration time is a relatively short time, e.g., three to twenty minutes, from the time at which the authentication token is created. In the embodiment shown, the expiration time is set at fifteen minutes from the time the authentication token is created. Note that it is important for the servers exchanging such authentication
20 tokens to maintain correct or synchronized clocks. The use of expiration time is used to create a single-use, perishable token.

In the embodiment shown, the authentication token built comprises a cookie with profile data of the customer 5 and an expiration time of fifteen minutes from the creation time of the token. Authentication tokens may also
25 comprise URL strings or other data that may be passed between servers. The profile data of the customer includes a customer identification number for the customer 5. The brokerage firm web server 30 includes a data storage system (e.g., a hard disk drive) that has customer identification numbers associated with log-in user names used by customer's of the brokerage firm web server 30. These
30 numbers were agreed upon previously by the administrators of the servers 30, 40. In the embodiment shown, a customer is associated with a customer identification number. In the embodiment, when the customer requests bill payment 50, the

server 30 retrieves, from the data storage system, the customer identification number for the customer 5. This customer identification number retrieved is used in the profile data used to build the cookie 52.

In another embodiment, various customer identification numbers are associated with various secondary servers. When the customer requests a service provided at a secondary site (or to be transferred to a secondary site), the primary server detects the request, determines the secondary site, and retrieves, from the data storage system, the customer identification number for the requesting customer that is associated with the secondary site to which the customer will be directed for bill payment services.

Referring again to FIG. 2, the server 30 also selects the secondary server recipient name 54. The server 30 does so by examining the request made by the customer and determining the name / address of the appropriate secondary server to handle the request. In the embodiment shown, the server 30 examines the "bill payment" request made by the customer. In the present embodiment, the examination comprises determining the Uniform Resource Locator (URL) associated with the "bill payment" hyperlink. The web page file associated with the welcome page 100 includes the URL associated with the "bill payment" hyperlink, and the server 30 selects this URL.

The server 30 then signs and encrypts the cookie 56. A digital signature associated with the brokerage firm server 30 is applied to the cookie 56 by the server 30. Preferably, the server 30 comprises public key encryption software capable of encrypting, digitally signing, and authenticating electronic transactions across applications and servers. In the embodiment shown, the Entrust/PKI 5.0 software package, with its associated application programming interface (API) library, available from Entrust Technologies, Inc., Plano, Texas, is used to sign the cookie using a public key encryption system. In the embodiment shown, the cipher used is the Triple DES (Data Encryption Standard) encryption algorithm system, the encrypted cookie uses privacy-enhanced mail (PEM) headers, and signing uses the Secure Hash Algorithm (SHA-1) to create the message digest (or hash value) in signing. The Triple DES system is used to encrypt the

authentication token (the cookie), and a PEM header and SHA-1 digest is included.

A digital signature associated with the server 30 (in the form of a signed-encrypted string) applied to the cookie allows a secondary server to verify that the authentication token was created by the brokerage firm server 30. Further, the signature allows a secondary server to detect any modification or corruption of the authentication token. The digital signature is applied and encrypted by the server 30.

In the embodiment shown, the cookie is created and kept on the browser 10 of the user 5 using a header entry in the response page, and the header entry is structured as follows: Set-cookie; name=value; expires=date-time; domain=domainname; path=path; secure. The path value comprises a specific directory, and the domainname value preferably is a common sub-domain (discussed further below). In an embodiment, the authentication token (the cookie) is non-persistent and will not be written to disk on the customer client 10. In such an embodiment, an expires value is not necessary, and the cookie will be deleted by the receiving server immediately after receipt.

An example of a string comprising cookie construction according to an embodiment of the present invention is as follows:

VER|1EXPDT|19990505132540||CT\CUST|AF|EXIST||CID|576001000560050234||
FCID|0||TA|2||ANM|0010000001||RTN|099||ANA|My wife's
Checking||AB|237600||ANM|0010000002||RTN|009||ANA|My
checking||AB|24556. The example is in the following format: Tag1 TagValue1
Tag2 TagValue2 Tag3 TagValue3

It is noted that the specific tags and their tag values may vary according to implementation needs, such as destination server requirements.

Tags in the example shown, their value, and a description is shown in the following table:

Tag	Tag Value	Description
VER	1	Version of cookie system being used (current version is 1)
EXPDT	Date / Time	The ASCII GMT date and time that the authentication token expires. Format: CCYYMMDDHHMMSS.
CT	CUST FC	Customer Type. CUST= regular customer. FC= Customer Representative. Used to activate view-only mode.
AF	NEW / EXIST	New or existing customer indicator.
CID	Integer	Customer Identification Number
FCID	Alphanumeric	Identification number for customer service representative. If the user is a regular customer, do not set FCID (i.e., set to 0).
TA	Integer	Total Number of Accounts of customer
ANM	Integer	Account Number
RTN	Integer	Routing transit Number – maps to prod-type-cd
ANA	Alphanumeric	Account nick name
AB	Integer	Account Balance in cents (i.e. \$10.50=1050)

- 5 In the embodiment shown, the order of tags is not significant, except for ANM, ANA, AB, which are considered a tuple and are grouped together as shown above. Also, in the embodiment shown, the VER, EXPDT, and CT are required tags. Other tags that may be used in other embodiments include the following: AG (agency or corporation name that employs the user), FNAM (first name of
- 10 user), and LNAM (last name of user).

The AF tag allows the bank server 40 to determine if an “Activate User” message should be sent to a transaction processing system (TPS). Preferably, if the brokerage firm web server 30 cannot determine whether a customer is new or

already enrolled with GTPS, the server 30 sets the AF tag to NEW. In addition, the brokerage firm web server 30 shown assumes that all accounts are of the "Checking" type, and will set the product type code based on the RTN value determined.

5 The brokerage firm server 30 then URL encodes (also called URLEncode) the constructed cookie 58. Essentially, in URL encoding the cookie, the broker firm web server converts the formatted string discussed above to a URL-encoded format. In one embodiment, the URL encoding encodes the string using the URL escape syntax which comprises a three character string (%nn) specifying the
10 hexadecimal code for a character. This syntax is used to hide characters that may be otherwise significant when used in a URL. The URL encoding 58 carried out by the brokerage firm web server 30 results in an encoded, signed, and encrypted string suitable for writing in a cookie, and such string is included in the cookie built by the server 30.

15 The brokerage firm web server 30 then sends a redirect command (or redirect page) with the URL encoded cookie (the authentication token) in a set-cookie header to the customer client 60. The redirect command includes the URL of the web site associated with "bill payment" 42. The customer client 10 receives the redirect command and the URL encoded cookie constructed by the brokerage
20 web server 30.

 The customer client 10 connects with the bank web site 42 via the Internet 20 and sends the cookie to the bank web server 62. In the embodiment shown, the authentication token is transmitted in a Secure Sockets Layer (SSL) session. Further, in the embodiment shown, on receipt of the redirect command, the
25 customer client 10 opens a second browser window, requests a download of the home page at the URL of the web site 42 (or the page designated in the URL), receives the page of the web site 42 from the bank web server 40, and displays the page in the window. An embodiment of such a window 110 and page is shown in FIG. 3. The cookie received by the customer client 10 from the brokerage firm
30 web server 30 is sent by the customer client 10 to the bank web server 40.

 The bank web server 40 receives the cookie from the customer client 10. The bank web server 40 decodes the encoded, signed, and encrypted string built

into the cookie by the brokerage firm web server 30 into a signed, encrypted string 64. Such decoding employs URL Decoding (or URLDecode) methods in the embodiment shown. In the embodiment shown, URL Decoding is employed to convert the URL encoded string in the cookie to plain ASCII for examination by the bank web server 40. The bank web server 40 and the brokerage firm web server 30 previously exchanged public-private key decryption information.

Once the string is decoded, the bank web server 40 decrypts and verifies the cookie (including the signed, encrypted string that is now decoded) 66. In the embodiment shown, software comprising public key encryption software capable of decrypting and authenticating electronic transactions across applications and servers is used by the bank web server 40 to do so. One example of such software is the Entrust/PKI 5.0 software package, with its associated application programming interface (API) library, available from Entrust Technologies, Inc., Plano, Texas.

The bank web server 40, using the software mentioned, determines whether the digital signature associated with the cookie verifies 68. If the signature does not verify, the bank web server 40 rejects the attempted sign-on and re-directs the customer client 10 to a web page on the brokerage firm web server 30 indicating an error and sign-on failure 70, resulting in a failed sign-on attempt 72. In another embodiment, if the signature does not verify, the bank web server 40 rejects the sign-on and sends a web page indicating an error and sign-on failure to the customer client 10. In an embodiment, if the signature does not verify, the bank web server 40 also sends a message indicating such failure to the brokerage firm web site 30, e.g., an e-mail message.

If the signature verifies, the bank web server 40 examines the Certificate Authority (CA) name associated with the cookie 74. The server 40 compares the CA name associated with the cookie (i.e., the CA used with the CA name expected, as recorded in a registry file in the bank web server 40). If the CA name is not the one expected, the bank web server 40 re-directs the customer client 10 to an error page on the brokerage firm web server 70 and the sign-on attempt fails 72, as described above.

If the CA name is the CA name expected, the bank web server 30 next if the sender's name is correct 76. In doing so, the bank web server 30 determines whether the Common Name (CN) associated with the cookie (i.e., the name being certified – the name used by the brokerage firm web server 30) is an authorized name. In so determining, the bank web server 40 compares the CN associated with the cookie with a file containing authorized names in a data registry in the bank web server 40. If the CN is not the one expected, the bank web server 40 re-directs the customer client 10 to an error page on the brokerage firm web server 70 and the sign-on attempt fails 72, as described above.

10 If the CN is correct, i.e., if the CN is an authorized name, the bank web server 40 extracts profile data from the cookie and begins a bill-payment session 78. In the embodiment shown, the server 40 parses the clear text data associated with the cookie (clear text refers to the information that is not encrypted) 78, and examines the expiration time data in the cookie (not shown). If the expiration time
15 has passed, the bank web server 40 re-directs the customer client 10 to an error page on the brokerage firm web server 70 and the sign-on attempt fails 72, as described above. The clear text data includes profile data (e.g., customer identification number). If the expiration time has not passed, the web server 40 begins a bill payment session using the session and profile data 78 by sending the
20 customer client the web page 110 of the bank web site 42 that is shown in FIG. 3, and the sign-on is successful 80. The customer client 10 receives the web page 100 and proceeds with the bill-payment session with the bank server 40. In an embodiment, the authentication token (cookie) is then discarded or destroyed by the web server 40.

25 In an alternative embodiment, the system reflects a service associated with a user's employee. One such embodiment is shown in FIG. 4. In the embodiment shown in FIG. 4, the process of this alternative embodiment generally proceeds as that discussed above, with exceptions discussed below. The embodiment shown employs a primary server comprising a central web server having a central web
30 site (not shown). The central web site comprises a web site at which the customer client 10 may request various services by clicking a hyperlink.

Referring to FIG. 4, the customer 5 signs on to the central web site 49, and the process 150, 152, 154, 156, 158, 160, 162, 164, 166, 168, 170, 174, continues in a manner like that described above in relation to steps 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 74 shown FIG. 3, with the central web server serving as the primary server (the brokerage web server served as the primary web server in the embodiment shown in FIG. 2) until the step shown as item 77 occurs. The primary server includes the following, additional tags in the cookie: AG (agency or corporation name that employs the user), FNAME (first name of user), and LNAME (last name of user). The service provider referred to in FIG. 4 comprises the secondary server, and in the embodiment shown comprises the bank web server 40. After the bank web server 40 determines that the sender's CA is correct, the web server 40 determines whether the customer / user's employer has signed up with the service provider associated with the secondary server (the bank web server 40) 77. The bank web server 40 includes a database containing a list of employers who signed up with the service offered by the bank web server 40. The web server 40 compares the agency or corporation name in the AG tag with the list of employers. If the name in the AG tag is not on the list, the web server 40 re-directs the customer client 10 back to the central web site for an error URL 70.

If the name in the AG tag is on the list, the web server 40 continues the process by extracting profile data from the cookie and beginning a bill-payment session 78. After the bank web server 40 extracts profile data from the cookie and begins a bill-payment session 78, the server 40 examines a database (not shown) associated with the server 40 that includes data reflecting users who have previously signed up for or used the service provided by the server 40. If the user reflected by the cookie exists (i.e., has previously signed up for or used the service provided by the server 40), the web server 40 retrieves a default web page previously selected as the user's default page and sends the default web page to the customer client 83, and the customer client 10 then may proceed with a bill-payment session 80.

If the user reflected by the cookie does not exist (i.e., the database does not reflect that the user has previously signed up for or used the service provided by

the server 40), the web server 40 creates a user identification using the tag information, including the AG, FNAM, and LNAM tag information, and stores the user identification in a database. The server 40 then retrieves a pre-designated default web page and transmits the web page to the customer client 83. The pre-designated default web page comprises a pre-designated web page for users associated with the agency / corporation indicated in the AG tag. The customer client 10 then may proceed with a bill-payment session 80.

In an embodiment, multiple secondary servers are used in an embodiment of the present invention. In still other embodiments, multiple primary servers are used. A group of one or more primary and one or more secondary servers sharing log-in and other information as described herein may be referred to as a "federation" of servers.

In the embodiments shown herein, a digital certificate generation software program is used to generate certificates. An example of such software is the Entrust Solo Version 4.0 software package, available from Entrust Technologies, Inc., Plano, Texas. Preferably, when multiple secondary servers are employed in an embodiment of the present invention, the secondary servers will use the same profile (a file comprising information needed by the certificate before the server can authenticate). Also, preferably, the CN name in the profile matches the generic host name of the secondary server.

In embodiments, multiple primary servers are employed. The primary servers may use the same profile on all hosts or each host may use a separate profile. Like the secondary servers, certificates are employed in each primary server.

In an embodiment, servers sharing information between themselves that are maintained by different organizations use a shared domain in common in order to ease the sharing of information. In such an embodiment, a sub-domain is established or designated as the common sub-domain, the domain attribute of the authentication token (e.g., the shared cookie) is designated as the common sub-domain, and a "Forward IP (Internet Protocol) Pointer" entry is added in the DNS name servers of the cooperating organizations.

For example, the primary server sets a cookie sub-domain of
 xxxx.yyyy.com (wherein yyyy.com comprises the domain name of the primary
 server). Using "tail matching," cookies may be shared with any host whose
 domain tail is "yyyy.com." When the server searches a cookie list on the user's
 5 computer for valid or useable cookies, the server compares the domain attributes
 of the cookie with the Internet domain name of the host. If there is a tail match,
 then the cookie will go through path matching to see if it should be sent. "Tail
 matching" means that domain attribute is matched against the tail of the fully
 qualified domain name of the host. For example, a domain attribute of
 10 "xxxx.com" would match host names "yyyy.xxxx.com" as well as
 "zzzz.yyyy.xxxx.com." For example, the primary server with which the user
 interacts has a domain name of qqqq.yyyy.com, and the secondary server has a
 domain name of ssss.rrrr.yyyy.com, may share cookies in such an embodiment. In
 an embodiment, the IP pointer in the domain name server associated with the
 15 primary server either (1) maps the secondary server domain (sss.rrrr.yyyy.com) to
 a pre-designated IP address associated with the secondary server; or (2) delegate
 the zone "rrrr" to a DNS name server associated with the secondary server for
 resolution.

As discussed above, certain embodiments of the present invention employ
 20 URL Encode and URL Decode. The following is a code fragment written in
 Microsoft C++ 6.0 showing an example of the implementation of URL Encode by
 a server:

```

//
// URLEncode converts some characters to %xx format for sending
// in a URL or writing to a cookie
//
void URLEncode (BYTE* szDecoded, BYTE* szEncoded)
{
    BYTE* pszInPtr = szDecoded;
    30    BYTE* pszOutPtr = szEncoded;
    BYTE inch;
```

```

while ((inch = *pszInPtr++) != '\0')
{
    if (    (inch < 32)    ||    (inch > 127)    ||
        (inch == '=') ||    (inch == '?') ||
5         (inch == '&') ||    (inch == '+') ||
        (inch == '%') ||    (inch == '-') ||
        (inch == ':') ||    (inch == ';') ||
        (inch == ','))
    {
10         *pszOutPtr++ = '%';
        sprintf((char*)pszOutPtr, "%02x", inch);
        pszOutPtr += 2;
    }
    else
15         *pszOutPtr++ = inch;
}
*pszOutPtr++ = '\0';
}

```

20 The following code fragment shows the implementation of URL-Decode
by as server:

```

//
// URLDecode converts characters in the %xx format back to their ascii
values
25 //
void URLDecode (BYTE* szEncoded, BYTE* szDecoded)
{
    BYTE* pszInPtr = szEncoded;
    BYTE* pszOutPtr = szDecoded;
30    int    inch;

    while ((inch = *pszInPtr++) != '\0')

```



```

{
    if (inch == '%')
        *pszOutPtr++ = (unHex(pszInPtr++) << 4) +
            unHex(*pszInPtr++);
5        else
            *pszOutPtr++ = inch;
    }
    *pszOutPtr++ = '\0';
}
10 int unHex (BYTE hexChar)
    {
        if ((hexChar >= '0') && (hexChar <= '9'))
            return hexChar - '0';
        if ((hexChar >= 'a') && (hexChar <= 'f'))
15            return hexChar - 'a' + 10;
        if ((hexChar >= 'A') && (hexChar <= 'F'))
            return hexChar - 'A' + 10;
        return 0;
    }
20
    Those of ordinary skill in the art will recognize that there are a variety of
    code fragments useful in carrying out such steps. Further, a variety of
    programming languages may be used.

    Various embodiments of the invention have been described in fulfillment
    of the various objects of the invention. It should be recognized that these
25    embodiments are merely illustrative of the principles of the present invention.
    Numerous modifications and adaptations thereof will be apparent to those skilled
    in the art without departing from the spirit and scope of the present invention.

```